

Univance API Gateway 1.0.0 OAS 3.0

Combined API documentation for all Univance microservices

Servers

http://localhost:3000/api - Local API Gateway server

Authorize



Authentication



POST

/auth/register Register a new user



GET

/auth/verify-email Verify user email



POST

/auth/resend-verification Resend email verification link



POST

/auth/login User login



POST

/auth/refresh-token Refresh access token



POST

/auth/forgot-password Request password reset



GET

/auth/reset-password Verify reset password token



POST

/auth/reset-password Reset user password



POST

/auth/logout User logout



PUT

/auth/update-password Update user password



GET

/auth/me Get current user profile



Attendance



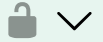
GET

/attendance Get all attendance records with filters



POST

/attendance Record attendance for a student



GET

/attendance/today Get today's attendance for a class



GET

/attendance/report Generate attendance report



POST

/attendance/bulk Record attendance for multiple students



GET

/attendance/{id} Get attendance records for a specific student



POST

/attendance/{id}/check-in Record daily check-in for a student



GET

/attendance/{id}/summary Get attendance summary with points



Badges



GET

/badges Get all badges



POST

/badges Create a new badge



GET

/badges/{id} Get badge by ID



POST

/badges/award Award a badge to a student



POST

/badges/check Check if student qualifies for badges



GET

/badges/criteria Get available badge criteria



GET

/badges/check-all/{studentId} Check all badge types for a student



GET

/badges/timeline/{studentId} Get achievement timeline for a student



GET

/badges/collections Get badge collections



POST

`/badges/collections` Create or update a badge collection

DELETE

`/badges/collections/{id}` Remove a badge from a collection

Link Requests



POST

`/link-requests/parent` Request to link with a parent

POST

`/link-requests/school` Request to link with a school

GET

`/link-requests/pending` Get pending link requests

DELETE

`/link-requests/{requestId}` Cancel a link request

GET

`/schools/join-requests` Get all pending join requests

POST

`/schools/join-requests/{requestId}` Respond to a join request

Parents



GET

`/parents/me` Get parent profile

GET

`/parents/me/children` Get all children

POST

`/parents/children` Add child (create or link)

POST

`/parents/profile` Create parent profile

PUT

`/parents/{id}` Update parent profile

POST

`/parents/link-code` Generate link code

DELETE

`/parents/children/{childId}` Remove child from parent

GET

`/parents/link-requests` Get pending link requests

POST

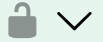
`/parents/link-requests/{requestId}` Respond to a link request

Schools

**POST****/schools** Create a new school**GET****/schools/me** Get current user's school profile**PUT****/schools/me** Update school profile**GET****/schools/me/teachers** Get all teachers for a school**POST****/schools/teachers** Add a teacher to the school**DELETE****/schools/teachers/{id}** Remove a teacher from the school**GET****/schools/me/students** Get all students for a school**GET****/schools/me/classes** Get all classes for a school**POST****/schools/import/students** Bulk import students**PUT****/schools/{id}/administrators** Update school administrators**POST****/schools/{id}/administrators** Add an administrator to a school**GET****/schools/{id}/administrators** Get all administrators for a school**DELETE****/schools/{id}/administrators/{userId}** Remove an administrator from a school**GET****/schools/join-requests** Get all pending join requests**POST****/schools/join-requests/{requestId}** Respond to a join request

Teachers

**GET****/schools/me/teachers** Get all teachers for a school**POST****/schools/teachers** Add a teacher to the school

DELETE**/schools/teachers/{id}** Remove a teacher from the school**GET****/teachers/me** Get current teacher profile**GET****/teachers/me/classes** Get classes for current teacher**POST****/teachers/profile** Create teacher profile**PUT****/teachers/{id}** Update teacher profile

Students

**GET****/schools/me/students** Get all students for a school**POST****/schools/import/students** Bulk import students**PATCH****/schools/{id}/points-account** Update student's points account ID**GET****/students/me** Get current student profile**GET****/students/{id}** Get student by ID**PUT****/students/{id}** Update student profile**POST****/students/profile** Create student profile**POST****/students/link/parent** Link with parent**GET****/students/{id}/badges** Get student badges**GET****/students/{id}/attendance** Get student attendance**POST****/students/{id}/check-in** Daily attendance check-in**GET****/students/{id}/level** Get student level info**POST****/students/link/school** Link with school**PATCH****/students/{id}/points-account** Update student's points account ID

DELETE**/students/{id}/parent/{parentId}** Unlink student from parent**DELETE****/students/{id}/school** Unlink student from school**POST****/students/request-parent** Request link with parent**POST****/students/request-school** Request link with school

Classes

**GET****/schools/me/classes** Get all classes for a school

School Classes

**GET****/classes/classes/{id}** Get class details**PUT****/classes/classes/{id}** Update a class**DELETE****/classes/classes/{id}** Delete a class**GET****/classes/classes/{id}/students** Get students in a class**POST****/classes/classes/{id}/students** Add a student to a class**POST****/classes/classes** Create a new class**POST****/classes/classes/{id}/join-code** Generate a new join code for a class**DELETE****/classes/classes/{id}/students/{studentId}** Remove a student from a class**GET****/classes/classes/{id}/join-requests** Get pending join requests for a class**POST****/classes/classes/{id}/join-requests/{requestId}** Respond to a join request

Search

**GET****/search/students** Search for students

GET

`/search/parents` Search for parents

GET

`/search/teachers` Search for teachers

GET

`/search/schools` Search for schools

Social Workers



GET

`/social-workers/me` Get current social worker profile

GET

`/social-workers/me/children` Get children assigned to current social worker

POST

`/social-workers/profile` Create social worker profile

PUT

`/social-workers/{id}` Update social worker profile

POST

`/social-workers/{id}/children` Assign child to social worker

DELETE

`/social-workers/{id}/children/{childId}` Remove child from social worker

POST

`/social-workers/{id}/case-notes` Add case note for a child

GET

`/social-workers/{id}/case-notes/{studentId}`
Get case notes for a specific child

Users



GET

`/users/me` Get current user profile

PUT

`/users/me` Update current user profile

PUT

`/users/me/password` Change user password

POST

`/users/me/avatar` Upload user avatar

GET

`/users/{id}` Get user by ID

Tasks



POST

`/tasks` Create a new task

GET

/tasks Get tasks with filtering

GET

/tasks/{id} Get a specific task by ID

PUT

/tasks/{id} Update a task

DELETE

/tasks/{id} Delete a task (soft delete)

POST

/tasks/{id}/complete Mark a task as completed

POST

/tasks/{id}/review Approve or reject a completed task

GET

/tasks/stats/analytics Get task statistics

GET

/tasks/student/{studentId}/summary Get tasks summary for a student

POST

/tasks/{id}/comments Add a comment to a task

POST

/tasks/{id}/generate-instance Generate next recurring task instance

Task Categories



POST

/tasks/categories Create a new task category

GET

/tasks/categories Get task categories with filtering

GET

/tasks/categories/{id} Get a specific task category by ID

PUT

/tasks/categories/{id} Update a task category

DELETE

/tasks/categories/{id} Delete a task category (soft delete)

POST

/tasks/categories/defaults Create default system categories

GET

/tasks/categories/context/{context} Get categories for a specific user context

Task Templates



POST

`/tasks/templates` Create a new task template

GET

`/tasks/templates` Get task templates with filtering

GET

`/tasks/templates/{id}` Get a specific task template by ID

PUT

`/tasks/templates/{id}` Update a task template

DELETE

`/tasks/templates/{id}` Delete a task template (soft delete)

POST

`/tasks/templates/{id}/use` Use a template to create a task

PUT

`/tasks/templates/{id}/featured` Toggle featured status for a template

POST

`/tasks/templates/system/defaults` Create default system templates

GET

`/tasks/templates/suggestions/{studentId}`
Get suggested templates for a student

Reward Redemptions



POST

`/rewards/redemptions/{id}/redeem` Redeem a reward

GET

`/rewards/redemptions` Get redemption history

GET

`/rewards/redemptions/pending` Get pending redemptions for fulfillment

GET

`/rewards/redemptions/{id}` Get redemption by ID

PUT

`/rewards/redemptions/{id}/fulfill` Fulfill a redemption

POST

`/rewards/redemptions/{id}/cancel` Cancel a redemption

Rewards



POST

`/rewards` Create a new reward

GET

`/rewards` Get available rewards with filtering

GET

`/rewards/{id}` Get reward by ID

PUT

`/rewards/{id}` Update a reward

DELETE

`/rewards/{id}` Delete a reward (soft delete)

POST

`/rewards/{id}/check-eligibility` Check if a reward can be redeemed

Reward Categories



POST

`/rewards/categories` Create a new reward category

GET

`/rewards/categories` Get reward categories with filtering

GET

`/rewards/categories/{id}` Get a specific reward category by ID

PUT

`/rewards/categories/{id}` Update a reward category

DELETE

`/rewards/categories/{id}` Delete a reward category (soft delete)

POST

`/rewards/categories/defaults` Create default system categories

GET

`/rewards/categories/hierarchy` Get hierarchical category structure

Point Accounts



POST

`/points/accounts` Create a new point account

GET

`/points/accounts/student/{studentId}` Get a student's point account details

GET

`/points/accounts/student/{studentId}/balance` Get a student's point balance

GET

`/points/accounts/student/{studentId}/level`
Get a student's level and progress information

GET

`/points/accounts/student/{studentId}/history`
Get a student's point transaction history

Point Configuration



GET

/points/configuration/active Get active point system configuration

GET

/points/configuration/history Get configuration version history

POST

/points/configuration Create new configuration version

GET

/points/configuration/{version} Get specific configuration version

POST

/points/configuration/{version}/activate
Activate a specific configuration version

GET

/points/configuration/levels Get all levels information

POST

/points/configuration/levels Add or update a level

DELETE

/points/configuration/levels/{level} Delete a level

Point Transactions



POST

/points/transactions Create a new point transaction (award or deduct points)

GET

/points/transactions/{id} Retrieve a point transaction by its ID

GET

/points/transactions/source/{source}/{sourceId}
Get transactions filtered by source and entity

GET

/points/transactions/student/{studentId}
List all transactions for a student with optional filters

GET

/points/transactions/student/{studentId}/summary
Get aggregated transaction summary for a student

POST

/points/transactions/{transactionId}/reverse Reverse a specific transaction

GET

/points/transactions/visualization/timeseries
Retrieve time series data for charting student points over time

GET

/points/transactions/analysis/categories
Analyze and summarize transactions by category

Analytics



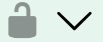
GET

/analytics/dashboard Get dashboard overview

GET

/analytics/health Get system health status

POST

/analytics/update Trigger manual analytics update

Badge Analytics



GET

/analytics/badges/awards Get badge award metrics

GET

/analytics/badges/popular Get popular badges

GET

/analytics/badges/categories Get badge category distribution

Points Analytics



GET

/analytics/points/transactions Get points transaction metrics

GET

/analytics/points/economy-health Get points economy health metrics

GET

/analytics/points/level-distribution Get level distribution metrics

Task Analytics



GET

/analytics/tasks/metrics Get task creation and completion metrics

GET

/analytics/tasks/categories Get task category distribution

GET

/analytics/tasks/schools-performance Get task performance by school

User Analytics



GET

/analytics/users/growth Get user growth metrics

GET

/analytics/users/engagement Get user engagement metrics

GET

/analytics/users/schools-comparison Get school comparison metrics

Notification Devices



POST

`/notifications/devices/register` Register a device for push notifications

GET

`/notifications/devices/me` Get user's devices

PUT

`/notifications/devices/{id}` Update a device

DELETE

`/notifications/devices/{id}` Delete a device

Notifications



POST

`/notifications/send` Send a notification to a user

POST

`/notifications/bulk` Send notifications to multiple users

GET

`/notifications/me` Get notifications for current user

GET

`/notifications/me/unread` Get unread notification count

POST

`/notifications/me/read/all` Mark all notifications as read

POST

`/notifications/{id}/read` Mark a notification as read

DELETE

`/notifications/{id}` Delete a notification

DELETE

`/notifications/me/all` Delete all notifications

GET

`/notifications/user/{userId}` Get notifications for a specific user

Notification Preferences



GET

`/notifications/preferences/me` Get current user's notification preferences

PUT

`/notifications/preferences/me` Update user's notification preferences

POST

`/notifications/preferences/me/reset`
Reset user's notification preferences to default

GET

`/notifications/preferences/user/{userId}` Get preferences for a specific user

POST

/notifications/preferences/role/{role}/defaults
Set default preferences for a role



Notification Templates



GET

/notifications/templates Get all notification templates



POST

/notifications/templates Create new notification template



GET

/notifications/templates/{id} Get template by ID



PUT

/notifications/templates/{id} Update notification template



DELETE

/notifications/templates/{id} Delete notification template



Schemas



Attendance >

Badge >

LinkRequest >

Parent >

Child >

School >

SchoolClass >

SocialWorker >

Student >

TeacherProfile >

User >

Task >

TaskCreate >

TaskCategory >

TaskCategoryCreate >

TaskTemplate >

TaskTemplateCreate >

TaskTemplateUpdate >

TaskFromTemplate >

RewardRedemption >

RedemptionRequest >

FulfillmentRequest >

CancellationRequest >

Reward >

RewardCreate >

RewardUpdate >

RewardEligibility >

RewardCategory >

RewardCategoryCreate >

RewardCategoryUpdate >

RewardCategoryHierarchy >

PointAccount >

PointConfiguration >

LevelInfo >

PointTransaction >

DashboardOverview >

SystemHealth >

AnalyticsUpdateRequest >

AnalyticsJob >

BadgeAwardMetricsResponse >

PopularBadgesResponse >

BadgeCategoryDistributionResponse >

PointsTransactionsResponse >

PointsEconomyHealthResponse >

LevelDistributionResponse >

TaskMetricsResponse >

TaskCategoryDistributionResponse >

SchoolPerformanceResponse >

UserGrowthResponse >

UserEngagementResponse >

SchoolComparisonResponse >

UserDevice >

DeviceRegister >

DeviceUpdate >

Notification >

NotificationSend >

NotificationBulkSend >

NotificationPreference >

PreferenceUpdate >

